Two-in-One-Depth: Bridging the Gap Between Monocular and Binocular Self-supervised Depth Estimation
Supplementary Material

Zhengming Zhou and Qiulei Dong ✉
State Key Laboratory of Multimodal Artificial Intelligence Systems, CASIA
School of Artificial Intelligence, UCAS
zhouzhengming2020@ia.ac.cn qldong@nlpr.ia.ac.cn

A. Multi-stage joint-training strategy

A.1. Image reconstruction

As mentioned in Sec. 3.4 of the main paper, the discrete depth constraint [1, 13, 2, 31] is used for helping TiO-Depth learn monocular depth estimation at step (1), which assumes that the depth of each pixel is inversely proportional to a weighted sum of a set of discrete disparities determined by the visual consistency between the input training stereo images [30]. A left-view reconstructed image $\hat{I}^l_a \in \mathbb{R}^{3\times H\times W}$ is obtained with the right-view real image $I^r \in \mathbb{R}^{3\times H\times W}$ and the predicted right-view auxiliary volume $V^r_a \in \mathbb{R}^{N\times H\times W}$ under the discrete depth constraint, where N is the number of the discrete disparity levels and $\{H, W\}$ are the height and width of the image. Specifically, a left-view auxiliary volume $\hat{V}^l_a \in \mathbb{R}^{N\times H\times W}$ is firstly generated by shifting the n^{th} channel of V^r_a with the corresponding disparity value b_n generated with the mirrored exponential disparity discretization [1]. Then, $\hat{V}^l_a$ is passed thought a softmax operation along the first dimension to obtain the corresponding probability volume $\hat{P}^l_a$. Accordingly, the left-view reconstructed image $\hat{I}^l_a$ is obtained by calculating a weighted sum of the shifted N versions of the right image I^r with $\hat{P}^l_a$:

$$\hat{I}^l = \sum_{n=0}^{N-1} \hat{P}^l_{an} \odot I^r_n \quad , \tag{1}$$

where $\hat{P}^l_{an} \in \mathbb{R}^{1\times H\times W}$ is the n^{th} channel of $\hat{P}^l_a$, '$\odot$' denotes the element-wise multiplication, and I^r_n is the left-view image shifted with b_n.

The continuous depth constraint [10, 11, 3] is used for helping TiO-Depth learn binocular depth estimation at step (2), which assumes that the depth of each pixel is a continuous variable determined by the visual consistency between the input training stereo images [30]. A left-view image $\tilde{I}^l_s$ is obtained with the right-view real image I^r and the predicted left-view depth map $D^l_s \in \mathbb{R}^{1\times H\times W}$ under the continuous depth constraint. Specifically, for an arbitrary pixel coordinate $p \in \mathbb{R}^2$ in the left-view image, its corresponding coordinate p' in the right image could be calculated with D^l_s:

$$p' = p - \left[\frac{Bf_x}{D^l_s(p)}, 0\right]^\top \quad , \tag{2}$$

where B is the baseline length of the stereo pair and f_x is the horizontal focal length of the camera. Accordingly, the reconstructed left-view image $\tilde{I}^l_s$ is obtained by assigning the RGB value of the right image pixel p' to the pixel p of $\tilde{I}^l_s$.

A.2. Monocular loss

The monocular loss used in step (1) contains a monocular reconstruction loss L_{rec1} and an edge-aware smoothness loss L_{smo1}. Specifically, L_{rec1} consists a L_1 loss term and a perceptual loss [18] term for measuring the similarity between the left-view reconstructed image $\hat{I}^l_a$ and the left-view real image I^l as done in [1, 31]:

$$L_{rec1} = \left\|\hat{I}^l_a - I^l\right\|_1 + \beta \sum_{i=1,2,3} \left\|\phi_i(\hat{I}^l_a) - \phi_i(I^l)\right\|_2 \quad , \tag{3}$$

where '$\|\cdot\|_1$' and '$\|\cdot\|_2$' denote the L_1 and L_2 norms, $\phi_i(\cdot)$ represents the output of i^{th} pooling layer of a pretrained VGG19 [26], and $\beta = 0.01$ is a balance parameter. The edge-aware smoothness loss L_{smo1} is employed for constraining the continuity of the auxiliary disparity map d^r_a as done in [10, 31, 1, 3]:

$$L_{smo1} = \|\partial_x d^r_a\|_1 e^{-\gamma\|\partial_x I^r\|_1} + \|\partial_y d^r_a\|_1 e^{-\gamma\|\partial_y I^r\|_1} \quad , \tag{4}$$

where '∂_x', '∂_y' are the differential operators in the horizontal and vertical directions respectively, and $\gamma = 2$ is a parameter for adjusting the degree of edge preservation.

Method	PP.	Sup.	Resolution	Abs Rel ↓	Sq Rel ↓	RMSE ↓	logRMSE ↓	A1 ↑	A2 ↑	A3 ↑
DepthHints [28]	✓	S(SGM)	320×1024	0.074	0.364	3.202	0.114	0.936	0.989	0.997
FAL-Net [12]	✓	S	384×1280	0.071	0.281	2.912	0.108	0.943	0.991	$\underline{0.998}$
PLADE-Net [13]	✓	S	384×1280	$\underline{0.066}$	0.272	2.918	0.104	0.945	0.992	$\underline{0.998}$
OCFD-Net [30]	✓	S	384×1280	0.069	0.262	2.785	0.103	0.951	$\underline{0.993}$	$\underline{0.998}$
SDFA-Net [31]	✓	S	384×1280	0.074	$\underline{0.228}$	**2.547**	0.101	0.956	**0.995**	**0.999**
TiO-Depth		S	384×1280	$\underline{0.066}$	0.229	2.597	$\underline{0.096}$	$\underline{0.961}$	**0.995**	**0.999**
TiO-Depth	✓	S	384×1280	**0.065**	**0.218**	$\underline{2.558}$	**0.094**	**0.962**	**0.995**	**0.999**
DepthFormer (2F.) [16]		M	320×1024	0.055	0.265	2.723	0.092	0.959	0.992	0.998
ManyDepth (2F.) [29]		M	352×1216	0.055	0.305	2.945	0.094	0.963	0.992	0.997
TiO-Depth (Bino.)		S	384×1280	**0.033**	**0.078**	**1.583**	**0.050**	**0.996**	**0.999**	**1.000**

Table A. Quantitative comparison on the improved KITTI Eigen test set. ↓ / ↑ denotes that lower / higher is better. The best and the second best results are in **bold** and underlined under each metric. The methods marked with '2F.' predict depths by taking 2 frames from a monocular video as input, while the methods with 'Bino.' predict depths by taking stereo pairs as input. 'PP.' means using the post-processing step. The methods marked with 'SGM' are trained with the the depth generated with SGM [17].

Method	train	test	Abs. Rel. ↓	Sq. Rel. ↓	RMSE ↓	logRMSE ↓	A1 ↑	A2 ↑	A3 ↑
PackNet [15]	D	D	0.173	7.164	14.363	0.249	0.835	-	-
ManyDepth (2F.) [29]	D	D	0.146	3.258	14.098	-	0.822	-	-
DepthFormer (2F.) [16]	D	D	**0.135**	2.953	**12.477**	-	**0.836**	-	-
TiO-Depth	K	D	0.144	**2.664**	14.273	**0.242**	0.808	0.933	0.970
MonoDepth2 [11]	C	C	0.129	1.569	6.876	0.187	0.849	0.957	0.983
Li *et al.* [20]	C	C	0.119	1.290	6.980	0.190	0.846	0.952	0.982
ManyDepth (2F.) [29]	C	C	**0.114**	1.193	6.223	0.170	**0.875**	**0.967**	0.989
SD-SSMDE [25]	C	C	**0.114**	**1.017**	**5.949**	**0.169**	0.870	**0.967**	**0.990**
MonoDepth2 [11]	K	C	0.153	1.785	8.590	0.234	0.774	0.926	0.976
SD-SSMDE [25]	K	C	0.143	1.635	8.441	0.221	0.789	0.931	0.980
TiO-Depth	K	C	**0.120**	**1.176**	**7.157**	**0.187**	**0.850**	**0.958**	**0.987**
TiO-Depth (Bino.)	K	C	0.066	0.423	4.070	0.106	0.961	0.992	0.997

Table B. Quantitative comparison on DDAD [15] and Cityscapes [4] (Tab. 3 in the main paper). 'C', 'K', and 'D' denote the methods are trained or tested on the Cityscapes, KITTI and DDAD datasets respectively.

A.3. Details of the training

Since the predicted depth results are not reliable at the early training epochs, which lack the ability to effectively guide the following steps, the second and third steps are enabled after $E_1 = 20$ and $E_2 = 30$ training epochs respectively. Thus, the multi-stage joint-training strategy contains three stages, where the training iterations are divided into one, two and three steps respectively as mentioned in Sec. 3.4 of the main paper. Considering that the second and the third steps are enabled after E_1 and E_2 epochs respectively and different parameters are optimized at these steps, we use three Adam optimizers [19] at the three steps for training. The learning rate of each optimizer is set to 10^{-4} when the corresponding training step is firstly enabled, and which is downgraded by half as described in Sec. 4.1 of the main paper. Since there are several parameters are trained only at one step (*e.g.*, the parameters in the monocular feature matching modules), while other parameters are trained at multiple steps (*e.g.*, the parameters in the decoder block), we multiply the learning rates of the parameters that have optimized at the previous steps by 0.1.

B. Dataset and metric

TiO-Depth is trained on the KITTI dataset [9] and evaluated on the KITTI, Cityscapes [4], and DDAD [15] datasets as mentioned in Sec. 4 of the main paper.

In addition to the Eigen split [7] and the KITTI 2015 stereo benchmark [23] which are employed for training and testing, an improved Eigen test set [27] comprised of 652 images with high-quality depth labels is also used for evaluation. The test set of Cityscapes [4] which contains 1525 stereo pairs with the disparity maps provided by SGM [17] and the validation set of DDAD which contains 3950 single images and the aligned LiDAR depth labels are used for evaluating the cross-dataset generalization ability of TiO-Depth,

The following seven metrics are used to evaluate the performances of monocular and binocular depth estimations on all the datasets:

- Abs Rel: $\frac{1}{N}\sum_i \frac{\left|\hat{D}_i - D_i^{gt}\right|}{D_i^{gt}}$
- Sq Rel: $\frac{1}{N}\sum_i \frac{\left|\hat{D}_i - D_i^{gt}\right|^2}{D_i^{gt}}$
- RMSE: $\sqrt{\frac{1}{N}\sum_i \left|\hat{D}_i - D_i^{gt}\right|^2}$

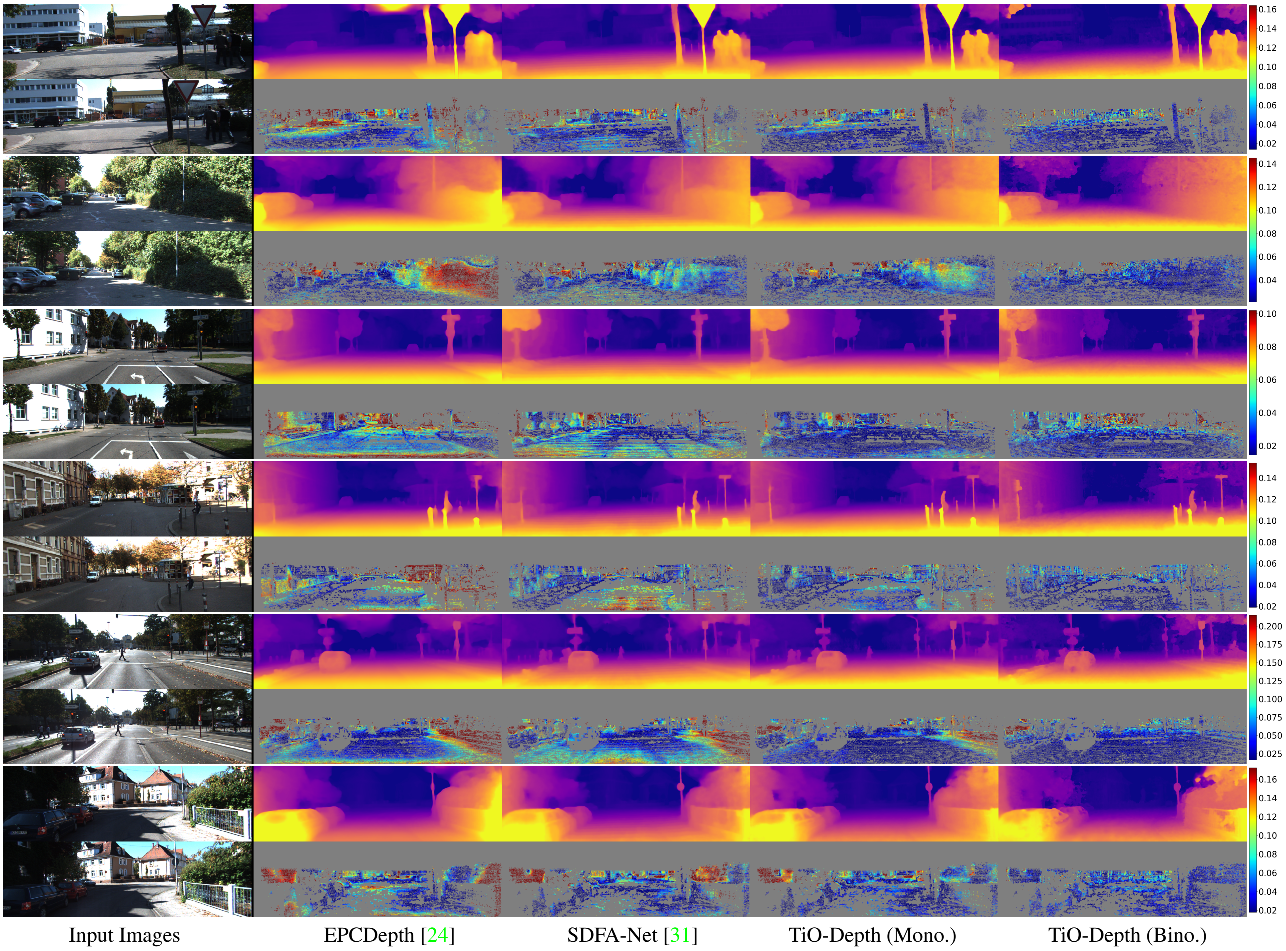

Figure A. Visualization results of EPCDepth [24], SDFA-Net [31] and our TiO-Depth on KITTI. The input stereo pairs are shown in the first column, where the left-view images are used for monocular depth estimation. The predicted depth maps with the corresponding 'Abs. Rel.' error maps calculated on the improved Eigen test set are shown in the following columns. For the error maps, red indicates larger error, and blue indicates smaller error as shown in the color bars.

- logRMSE: $\sqrt{\frac{1}{N}\sum_i \left|\log\left(\hat{D}_i\right) - \log\left(D_i^{gt}\right)\right|^2}$
- Threshold (Aj): % $\quad s.t. \quad \max\left(\frac{\hat{D}_i}{D_i^{gt}}, \frac{D_i^{gt}}{\hat{D}_i}\right) < a^j$

where $\{\hat{D}_i, D_i^{gt}\}$ are the predicted depth and the ground-truth depth at pixel i, and N denotes the total number of the pixels with the ground truth. In practice, we use $a^j = 1.25, 1.25^2, 1.25^3$, which are denoted as A1, A2, and A3 in all the tables. EPE and D1 metrics are also adopted for the evaluation of binocular depth estimation as done in [21, 22]:

- EPE: $\frac{1}{N}\sum_i \left|\hat{d}_i - d_i^{gt}\right|$
- D1: % $\quad s.t. \left(\left|\hat{d}_i - d_i^{gt}\right| > 3\right) \vee \left(\frac{\left|\hat{d}_i - d_i^{gt}\right|}{d_i^{gt}} > 0.05\right)$

where $\{\hat{d}_i, d_i^{gt}\}$ are the predicted disparity and the ground-truth disparity at pixel i.

For the evaluation on the raw and improved KITTI Eigen test sets [7, 27], we use the center crop proposed in [8] and the standard cap of 80m. For the evaluation on the KITTI 2015 training set, all the ground truth disparities are used for calculating D1 and EPE metrics, while other metrics are calculated with the cap of 80m as done in [3]. For the evaluation on the DDAD dataset [15], the cap of 200m is used, while the input images are resized into the resolution of 384×640 as done in [15]. For the evaluation on the Cityscapes dataset [4], we use the center crop and the standard cap of 80m as done in [29, 14, 20], while the input images are cropped and resized into the resolution of 192×512 as done in [29]. All the cross-dataset results of TiO-Depth are calculated after the median scaling [6].

Methods	Abs. Rel. ↓	Sq. Rel. ↓	RMSE ↓	logRMSE ↓	A1 ↑	A2 ↑	A3 ↑	EPE ↓	D1 ↓
w. Cat module (321)	0.069	0.505	3.442	0.123	0.947	0.983	0.992	2.074	15.952
w. Attn module (321)	0.053	0.439	3.214	0.106	0.965	0.987	**0.994**	1.377	7.421
w. MFM (1)	0.054	**0.423**	3.211	0.109	0.960	0.986	0.993	1.483	8.784
w. MFM (21)	0.052	0.445	3.268	0.107	0.965	0.987	**0.994**	1.305	7.077
TIO-Depth	**0.051**	0.429	**3.137**	**0.105**	**0.966**	**0.988**	**0.994**	**1.281**	**6.684**
w/o. L_{gui}	0.053	0.506	3.378	0.108	**0.966**	0.987	0.993	1.292	6.984
w/o. L_{gui}, L_{cos}	0.053	0.522	3.404	0.110	0.965	0.986	0.993	1.326	6.775
w/o. L_{gui}, L_{cos}, M_{occ}	0.054	0.565	3.637	0.121	0.963	0.984	0.992	1.345	7.159

Table C. Binocular depth estimation results on KITTI 2015 training set in the ablation study (Tab. 4 in the main paper). The numbers in the name of methods mean the indexes of the used modules as shown in Fig. 2 of the main paper. All the results are evaluated after training 30 epochs.

Steps	L_{dis}	FB.	Abs. Rel. ↓	Sq. Rel. ↓	RMSE ↓	logRMSE ↓	A1 ↑	A2 ↑	A3 ↑
1	-	-	0.088	0.556	4.093	0.173	0.904	0.967	0.984
1+2	-	-	0.088	0.557	4.067	0.172	0.906	0.968	0.984
1+2+3	P_s^l	✓	0.086	0.590	4.021	0.169	0.911	0.969	0.985
1+2+3	P_h^l	✓	**0.085**	**0.544**	**3.919**	**0.169**	**0.911**	**0.969**	**0.985**
1+2+3	P_h^l	-	0.098	0.695	4.367	0.183	0.892	0.964	0.983

Table D. Monocular depth estimation results predicted by TiO-Depth on the KITTI Eigen test set in the ablation study (Tab. 5 in the main paper). 'FB.' denotes using the final branches.

C. Comparative Results

As done in [28, 12, 13, 30, 31], we evaluate TiO-Depth on the improved KITTI Eigen test set [27] and the corresponding results are shown in Tab. A. It can be seen that TiO-Depth outperforms all the comparative methods in most cases in both monocular and binocular (multi-frame) tasks. Additional visualization results are given in Fig. A. These results further demonstrate the effectiveness of TiO-Depth as a two-in-one model.

In Tab. B, the monocular and binocular depth estimation results of TiO-Depth and 6 comparison methods [20, 11, 15, 16, 25, 29] on the DDAD [15] and Cityscapes [4] datasets under all the seven metrics are given, which demonstrate the generalization ability of TiO-Depth on the unseen datasets.

D. Ablation Study

We have verified the effectiveness of each key element in TiO-Depth by conducting ablation studies on the KITTI dataset [9] in Sec. 4.3 of the main paper. Tab. C shows the binocular depth estimation results in the ablation study under all of the nine metrics, which demonstrate the effectiveness of the dual-path decoder and the stereo loss L_S on the binocular task.

The monocular depth estimation results in the ablation study under all of the seven metrics are shown in Tab. D, which indicate the effectiveness of the multi-stage joint-training strategy. Furthermore, the results also prove the significance of the final branches in the Self-Distilled Feature Aggregation (SDFA) [31] blocks (as shown in Fig. B(a) where the raw data path in blue is used as the auxiliary branch and the distilled branch in red is used as the final branch) for the monocular task.

To further explore the effect of such switchable branches on learning more accurate monocular depths, a variant of TiO-Depth is built by replacing the three SDFA blocks in the dual-path decoder by the switchable aggregation blocks shown in Fig. B(b). The switchable aggregation block is inspired by the deformable convolution [5, 32] and is built based on the basic decoder block described in Sec. 3.2 of the main paper. In comparison to the basic decoder block, it employs two additional 3 × 3 convolutional layers as the switchable 'final branches' to learn the spatial offsets for the kernels of the convolutional layers in the basic decoder block. Accordingly, the standard convolutional layers in the basic block are converted to the deformable convolutions when the final branches are used. We train this variant with the multi-stage joint-training strategy and conduct the ablation studies. The corresponding results are shown in Tab. E. It can be seen that the whole performances of the variant TiO-Depth are poorer than that of TiO-Depth shown in Tab. D, mainly because the SDFA blocks could aggregate the features more effectively than the basic decoder layers. However, using the switchable final branches significantly improves the performance of the model in comparison to that without the final branches. These results further demonstrate that the potential of TiO-Depth for employing a more general architecture.

Finally, we conduct the ablation study on the input image resolution. As seen from Tab. F, TiO-Depth still performs well under the two low resolutions.

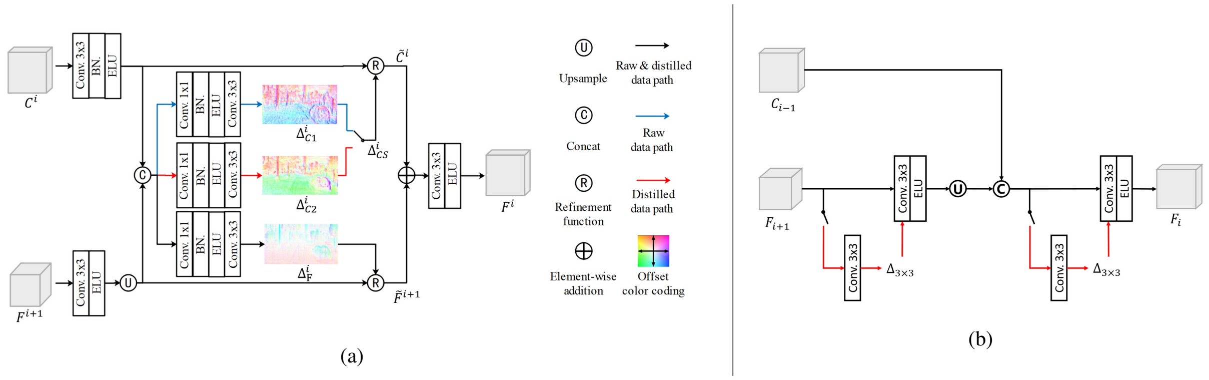

Figure B. (a) Architecture of the Self-Distilled Feature Aggregation (SDFA) block cited from [31]. (b) Architecture of the switchable feature aggregation block inspired by the deformable convolution [5, 32].

Steps	L_{dis}	FB.	Abs. Rel. ↓	Sq. Rel. ↓	RMSE ↓	logRMSE ↓	A1 ↑	A2 ↑	A3 ↑
1	-	-	0.094	0.579	4.155	0.178	0.896	0.966	0.984
1+2	-	-	0.094	0.582	4.165	0.177	0.896	0.966	0.984
1+2+3	P_h^l	✓	**0.086**	**0.551**	**3.967**	**0.170**	**0.907**	**0.969**	**0.985**
1+2+3	P_h^l	-	0.103	0.688	4.367	0.181	0.890	0.966	0.984

Table E. Monocular depth estimation results predicted by the variant of TiO-Depth on the KITTI Eigen test set in the ablation study.

References

[1] Juan Luis Gonzalez Bello and Munchurl Kim. Forget about the lidar: Self-supervised depth estimators with med probability volumes. *Advances in Neural Information Processing Systems*, 33, 2020. 1

[2] Juan Luis Gonzalez Bello and Munchurl Kim. Self-supervised deep monocular depth estimation with ambiguity boosting. *IEEE TPAMI*, 2021. 1

[3] Zhi Chen, Xiaoqing Ye, Wei Yang, Zhenbo Xu, Xiao Tan, Zhikang Zou, Errui Ding, Xinming Zhang, and Liusheng Huang. Revealing the reciprocal relations between self-supervised stereo and monocular depth estimation. In *ICCV*, pages 15529–15538, 2021. 1, 3

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2, 3, 4

[5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 4, 5

[6] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, pages 2650–2658, 2015. 3

[7] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 2, 3

[8] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, pages 740–756, 2016. 3

[9] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, pages 3354–3361, 2012. 2, 4

[10] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, pages 270–279, 2017. 1

[11] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, pages 3828–3838, 2019. 1, 2, 4

[12] Juan Luis GonzalezBello and Munchurl Kim. Forget about the lidar: Self-supervised depth estimators with med probability volumes. *Advances in Neural Information Processing Systems*, 33:12626–12637, 2020. 2, 4

[13] Juan Luis GonzalezBello and Munchurl Kim. Plade-net: Towards pixel-level accuracy for self-supervised single-view depth estimation with neural positional encoding and distilled matting loss. In *CVPR*, pages 6851–6860, 2021. 1, 2, 4

[14] Ariel Gordon, Hanhan Li, Rico Jonschkowski, and Anelia Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8977–8986, 2019. 3

[15] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *CVPR*, pages 2485–2494, 2020. 2, 3, 4

[16] Vitor Guizilini, Rareș Ambruș, Dian Chen, Sergey Zakharov, and Adrien Gaidon. Multi-frame self-supervised depth with transformers. In *CVPR*, pages 160–170, 2022. 2, 4

Method	Resolution	Abs Rel↓	Sq Rel↓	RMSE↓	logRMSE↓	A1↑	A2↑	A3↑
TiO-Depth	192×640	0.091	0.625	4.179	0.174	0.902	0.968	0.984
TiO-Depth	320×1024	0.087	0.566	3.970	0.170	0.910	0.969	0.985
TiO-Depth	384×1280	0.085	0.544	3.919	0.169	0.911	0.969	0.985
TiO-Depth (Bino.)	192×640	0.065	0.572	3.767	0.157	0.940	0.971	0.984
TiO-Depth (Bino.)	320×1024	0.064	0.526	3.594	0.153	0.943	0.973	0.985
TiO-Depth (Bino.)	384×1280	0.063	0.523	3.611	0.153	0.943	0.972	0.985

Table F. Depth estimation results with different input image resolutions on the KITTI Eigen test set in the ablation study.

[17] Heiko Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *CVPR*, volume 2, pages 807–814. IEEE, 2005. 2

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711, 2016. 1

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[20] Hanhan Li, Ariel Gordon, Hang Zhao, Vincent Casser, and Anelia Angelova. Unsupervised monocular depth learning in dynamic scenes. In *Conference on Robot Learning*, pages 1908–1917. PMLR, 2021. 2, 3, 4

[21] Liang Liu, Guangyao Zhai, Wenlong Ye, and Yong Liu. Unsupervised learning of scene flow estimation fusing with local rigidity. In *IJCAI*, pages 876–882, 2019. 3

[22] Pengpeng Liu, Irwin King, Michael R Lyu, and Jia Xu. Flow2stereo: Effective self-supervised learning of optical flow and stereo matching. In *CVPR*, pages 6648–6657, 2020. 3

[23] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. In *ISPRS Workshop on Image Sequence Analysis (ISA)*, 2015. 2

[24] Rui Peng, Ronggang Wang, Yawen Lai, Luyang Tang, and Yangang Cai. Excavating the potential capacity of self-supervised monocular depth estimation. In *ICCV*, pages 15560–15569, 2021. 3

[25] Andra Petrovai and Sergiu Nedevschi. Exploiting pseudo labels in a self-supervised learning framework for improved monocular depth estimation. In *CVPR*, pages 1578–1588, 2022. 2, 4

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[27] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 international conference on 3D Vision (3DV)*, pages 11–20, 2017. 2, 3, 4

[28] Jamie Watson, Michael Firman, Gabriel J Brostow, and Daniyar Turmukhambetov. Self-supervised monocular depth kints. In *ICCV*, pages 2162–2171, 2019. 2, 4

[29] Jamie Watson, Oisin Mac Aodha, Victor Prisacariu, Gabriel Brostow, and Michael Firman. The temporal opportunist: Self-supervised multi-frame monocular depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1164–1174, 2021. 2, 3, 4

[30] Zhengming Zhou and Qiulei Dong. Learning occlusion-aware coarse-to-fine depth map for self-supervised monocular depth estimation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6386—-6395, 2022. 1, 2, 4

[31] Zhengming Zhou and Qiulei Dong. Self-distilled feature aggregation for self-supervised monocular depth estimation. In *ECCV*, pages 709–726. Springer, 2022. 1, 2, 3, 4, 5

[32] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9308–9316, 2019. 4, 5